



TITLE:

分散システムの一つのスケジューリング問題(アルゴリズムの数学的基礎理論とその応用)

AUTHOR(S):

阿江, 忠; 山下, 雅史; 方, 安祥

CITATION:

阿江, 忠 ...[et al]. 分散システムの一つのスケジューリング問題(アルゴリズムの数学的基礎理論とその応用). 数理解析研究所講究録 1986, 591: 105-110

ISSUE DATE:

1986-05

URL:

<http://hdl.handle.net/2433/99473>

RIGHT:

分散システムの一つのスケジューリング問題

広島大学工学部 阿江 忠 (Tadashi Ae)

広島大学工学部 山下雅史 (Masafumi Yamashita)

広島大学工学部 方 安祥 (An Xiang Fang)

1. まえがき

分散システムの一つである LAN の方式についてはいろいろ議論されてきたが、現在では、CSMA/CD 方式あるいはトークンパッシング方式に大別できる。ここではトークンパッシング方式 (図 1) における一つのスケジューリング問題をモデル化し、その計算複雑さを論じる。

通常のトークンパッシング方式では、図 2 (a) のようにトークンをつかまえたプロセッサの処理時間は可変である (上限はパケット長に、下限は実現方法に依存する)。ここでは 理論的な取り扱いを容易にするため、各プロセッサにおけるトークンの滞在時間を一定としたスロット化トークンパッシングのモデルを提案する。

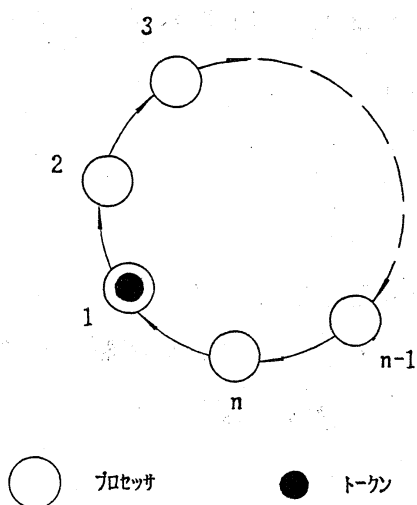
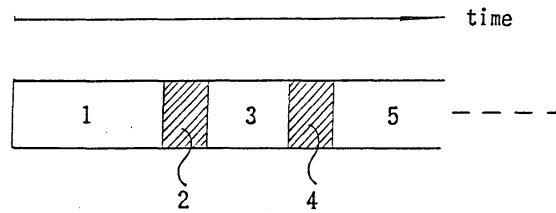
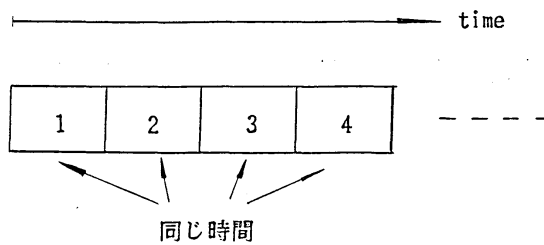


図 1 . トークンパッシング方式



(a) 通常の場合



(b) スロット化トークンパッシング

図2. モデリング

このスロット化トークンパッシング・モデルにおけるスケジューリング問題の計算複雑さは、各プロセスに属するタスク間の先行関係が線形順序であるという制約のもとでも一般にはNP-完全になることを示す。

2. スロット化トークンパッシング・スキーマのスケジューリング

スロット化トークンパッシング・スキーマのスケジューリング問題（以下では単にスケジューリング問題という）は以下のように定義される。

スケジューリング問題：

インスタンス：スロット化タスク（以下，単にタスク）の集合

$\mathcal{T} = \{t_1, t_2, \dots, t_n\}$ ，ただし t_i ($i = 1, 2, \dots, n$) はプロセッサ i でのみ実行可能としプロセッサは $1, 2, \dots, n$ ， $1, 2, \dots, n, \dots$ の順に実行できる。

線形順序の制約を持つ m 個のジョブ J_1, \dots, J_m ，ただし

$$J_i = t_{i,1} t_{i,2} \dots t_{i,j} \\ (t_{i,j} \in \mathcal{T}, t_{i,j} \neq t_{i,k} \text{ if } j \neq k)$$

とする。

質問： $t_1 t_2 \dots t_n$ までの一巡をラウンドと呼ぶ時ラウンド数 b 以内ですべての J_1, \dots, J_m を終了するスケジュールが存在するか？
すなわち、以下の条件を満足するスケジュール

$$\sigma : \{t_{i,j}\} \rightarrow \mathbf{N} \quad (\mathbf{N} : \text{自然数})$$

は存在するか？

条件： (i) σ : 単射

$$(ii) \quad \forall i, j \quad \sigma(t_{i,j}) < \sigma(t_{i,j+1})$$

$$(iii) \quad \forall i, j \quad t_{i,j} = t_{(\sigma(t_{i,j}) - 1) \bmod n + 1}$$

$$(iv) \quad \max \{ \sigma(t_{i,j}) \} \leq bn$$

3. NP-完全の問題

<場合 1> $|\mathcal{T}| = 2$ でもスケジューリング問題はNP-完全である。

(略証) $\mathcal{T} = \{1, 2\}$ とする。スケジューリング問題がNPであることは明らか。分割問題(Partition問題) がスケジューリング問題に多項式時間変換可能であることを示す。

分割問題のインスタンスを $A = \{a_1, \dots, a_n\}$ とするとき以下のようにスケジューリング問題を構成する。ここに、

$$\sum_{i=1}^n a_i = 2b \quad (b: \text{自然数})$$

と一般性を失わずに仮定できる。

次のように n 個のジョブ J_0, J_1, \dots, J_n を作る。

$$J_0 = 2 \ 2b \ 1 \ 2b \ 2 \ 2b \ 1 \ 2b$$

$$J_1 = 1 \ 2a_1 \ 2 \ 2a_1$$

$$J_2 = 1 \ 2a_2 \ 2 \ 2a_2$$

.

.

.

$$J_n = 1 \ 2a_n \ 2 \ 2a_n$$

まず、上記のジョブに $8b$ ラウンドのスケジュールが存在するとき、 $A' \subset A$ が存在して $\sum_{a \in A'} a = b$ なる A' なる分割の存在することを言う。上記の作り方から、 $8b$ ラウンドのスケジューリングが存在するためには、必ず J_0 のなかの1つの"2"と $J_1 \sim J_n$ のなかの"1"つの1を組み合わせる1つのラウンドを作り、 J_0 のなかの1つの"1"と $J_1 \sim J_n$ のなかの1つの"2"を組み合わせる1つのラウンドを作らなければならないことが証明でき、これを用いれば、容易に示すことができる。

次に分割の解 $A' \subset A$ が存在するときは、必ず J_i ($i=1, \dots, n$) と J_0 を組み合わせた $8b$ ラウンドのスケジュールが存在することを言う。 $A' = \{a_1, \dots, a_j\}$ を一般性を失うことなく仮定する。以下のスケジュールを考える。

最初 J_1, \dots, J_j の先頭の1の列と J_0 の先頭の $2b$ 個の2の列を組み合わせる。次に、 J_0 の前の $2b$ 個の1の列を J_1, \dots, J_j の残りの2の列と組み合わせる。更に、 J_{j+1}, \dots, J_n の先頭の1と J_0 の残つ

た $2b$ 個の 2 の列を組み合わせ、最後に、 J_0 の残った 1 の列と J_{j+1}, \dots, J_n の 2 の列を組み合わせる。このことにより、 $8b$ のスケジュールが構成される。

以上から分割問題はこのスケジューリング問題に変換することができるので、このスケジューリング問題は NP-完全である。 \square

〈場合 2〉すべてのジョブが単調なタスクシーケンスであっても NP-完全である。ここに、ジョブ $J_i = t_{i1} \dots t_{i_j}$ が単調であることは、任意の p, q ($p < q$) に対し、 $t_{ip} = t_{i'p}$, $t_{iq} = t_{iq'}$ とすれば、 $p' < q'$ を満たすことである。

(略証) 独立節点集合による節点の被覆問題^[1]をスケジューリング問題に帰着する。

具体的には問題『3つの独立節点集合で節点被覆できるか?』が NP-完全になることを使って証明をする。

グラフ $G = (V, E)$,

$$V = \{v_1, \dots, v_p\}, E = \{e_1, \dots, e_q\}$$

に対して

$$\mathcal{J} = \{v_1, \dots, v_p, e_1, \dots, e_q, w_1, \dots, w_p\}$$

ただしタスクは左から順にオーダーがついているものとする。

$v_i \in V$ に対して

$$E v_i = \{f_1, \dots, f_j, \dots, f_r \mid f_j = (u, v) \in E\}$$

とする。

各 v_i に対してつぎの3つのジョブを作る。

$$J^1_{v_i} = v_i f_1 \dots f_r w_i$$

$$J^2_{v_i} = v_i w_i$$

$$J^3_{v_i} = v_i w_i$$

3 | V | 個のジョブに対して、3 ラウンドのスケジュールが存在する必要十分条件は3つの独立集合による被覆が存在することである。必要条件は明らか。十分条件は v_i と w_i を各ジョブにつけてあることに注意すれば証明できる。3つの独立集合の被覆問題をすべてのジョブが単調なタスキシーケンスであるスケジューリング問題に変換することができるのでこのスケジューリング問題はNP-完全である。 \square

4. むすび

一見簡単に見えるスロット化トークンパッシング方式上のスケジューリング問題も一般にはNP-完全なることが判明した。最後に多項式時間で解ける部分問題を示す。

〈場合 3〉 ジョブの数を c (定数), シーケンスの最大の長さを l とすると、DPを用いて $O(l^c)$ で解ける。

文献.

- [1] Garey, M.R. and Johnson, D.S., "Computers and Intractability: A Guide to the Theory of NP-Completeness", W.H. Freeman and Company, 1979
- [2] Ullman, J.D., "NP-Complete Scheduling Problems", JCSS, vol. 10, pp. 384-393, 1975